



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicant : Wolrich, et. al. Art Unit : 2143
Serial No.: 09/473,571 Examiner : England, David
Filed : 12/28/1999 Assignee : Intel Corporation
Title : PROVIDING REAL-TIME CONTROL DATA FOR A NETWORK
PROCESSOR

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

(i)	Real Party in Interest	...page 2.
(ii)	Related Appeals and Interferences	...page 2.
(iii)	Status of Claims	...page 2.
(iv)	Status of Amendments	...page 2.
(v)	Summary of Claimed Subject Matter	...page 3.
(vi)	Grounds of Rejection to be Reviewed on Appeal	...page 5.
(vii).	Arguments	...page 5.
(viii)	Claims Appendix	...page 11.

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

I hereby certify under 37 CFR §1.8(a) that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

01/20/2006 BABRAHA1 00000079 09473571

~~01 FC:1251~~ ~~120.00 DP~~
02 FC:1402 120.00 DA 380.00 DP

January 13, 2006

Date of Deposit


Signature

Denise Donahue

Typed or Printed Name of Person Signing Certificate

(i) Real Party in Interest

The real party in interest in this appeal is Intel Corporation, a Delaware corporation having a principal place of business at 2200 Mission College Blvd, Santa Clara, CA 95052. Intel is the assignee of the entire right, title, and interest in the above-noted application.

(ii) Related Appeals and Interferences
None.

(iii) Status of Claims

Claims 1-40 are pending, stand rejected, and are being appealed with claims 1, 9, 18, 28, and 33 being independent.

(iv) Status of Amendments

After the Final Office Action mailed 07/13/2005, Applicants presented amendments to dependent claims 3, 6, 7, 8, 10, 14, 21, 22, 23 to clarify that the antecedent basis for the recited "device" or "devices" in these claims is provided by the recitation of "media access" device(s) in the corresponding independent claims. The Examiner denied entry of these amendments indicating they "raise new issues that would require further search and consideration".

Applicants also presented an amendment to claim 5 to fix a typographical error (i.e., correct the tense of "schedule" to "scheduled"). The Examiner did not enter this amendment indicating that it raised new issues that would require further search and consideration.

Finally, Applicants presented amendments to dependent claims 38 and 39 in response to a request for clarification in the Final Office Action. The Examiner did not enter these amendments, again, indicating they also raised new issues that would require further search and consideration.

(v) Summary of Claimed Subject Matter

A. Claim 1

Claim 1 recites a processor (e.g., FIG. 1, item 12; page 3, line 22 - page 4, line 1). The processor includes a module (e.g., FIG. 2, item 50; page 6, lines 6-11) configured to collect status data from media access devices (e.g., FIG. 1, items 14, 14', 14"; page 3, line 7-line 8) connected to a bus. The status data indicates readiness of the media access devices to participate in data transfers and includes data indicating whether a one of the media access devices has received packet data (e.g., page 10, line 18 - page 11, line 3). The processor also includes one or more processing engines (e.g., FIG. 1, items 22a-22f; page 3, lines 21-23; FIG. 3; page 9, lines 1-16) to schedule transfers of packet data (page 6, line 18 - page 7, line 5). The processor further includes a push engine (e.g., FIG. page 10, line 18 - page 11, line 32, item 62; page 6, lines 14-18) to perform unsolicited transfers of the status data to the processing engines in response to the module collecting new status data (FIG. 6, page 13, line 12 - page 14, line 6).

Claims 9 and 28 are method and computer-readable medium claims that share similar limitations. In particular, claim 9 recites transferring data packets (e.g., page 3, lines 14-16) over a bus. The method includes collecting information on readiness of media access devices (e.g., FIG. 6, item 102) connected to the bus to one of transmit and receive data packets (e.g., page 6, lines 12-14) and transferring a portion of the collected information to a processing engine configured to schedule data transfers where the transferring is unsolicited by the processing engine (e.g., FIG. 6, page 13, line 12 - page 14, line 6).

C. Claim 18

Claim 18 recites a router (e.g., FIG. 1; page 3, lines 2-12) that includes a bus and a parallel processor (e.g., FIG. 1, item 12) coupled to the bus. The parallel processor

includes a plurality of processing engines (e.g., FIG. 1, items 22a-22f; FIG. 3; page 9, lines 1-16) to process data transfers (e.g., page 3, lines 14-16) with a plurality of media access devices (e.g., FIG. 1, items 14, 14', 14"; page 3, lines 7-8) connected to the bus. The parallel processor also includes an interface (e.g., FIG. 1, item 38; page 5, lines 13-18) connected to collect ready status data from the media access devices and to automatically transfer ready status data to the processing engines in response to the status data being collected (FIG. 6; page 13, line 12 - page 14, line 6). The ready status data indicating readiness of the devices to participate in data transfers and includes data indicating whether a one of the media access devices has received packet data (page 10, line 18 - page 11, line 3).

D. Claim 33

Claim 33 recites a processor (e.g., FIG. 1, item 12; page 3, line 22 - page 4, line 1) that includes multiple multi-threaded programmable processing engines (e.g., FIG. 1, items 22a-22f; page 3, line 22 - page 4, line 1; FIG. 3; page 9, lines 1-16), individual ones of the programmable processing engines having at least one register (e.g., FIG. 3, item 78; page 9, lines 10-12). The processor also includes an interface (FIG. 1, item 38) operationally coupled to the multiple programmable processing engines. The interface includes at least one register (e.g., FIG 2, item 54; page 6, lines 11-12. The interface includes logic (FIG. 2, item 50; page 6, lines 9-11) to collect status data of at least one media access device via a bus the status data indicating whether the at least one media access device has received packet data. The interface also includes logic (FIG. 2, item 62) to perform a transfer, unsolicited by the programmable processing engines, of at least a portion of the collected status data stored in the at least one register of the interface to at least one register of the multiple multi-threaded programmable processing engines (page 13, line 12 - page 14, line 6).

(vi) Grounds of Rejection to be Reviewed on Appeal

Claims 1-5, 7-11, 13, 14, 16, 17, and 33-40 stand rejected under 35 U.S.C. 103(a) as obvious over Isfeld (U.S. Pat. No. 5,592,622) in view of Chilton (U.S. Pat. No. 6,418,488) in further view of Witkowski (6,430,626).

Claims 18, 19, 22, and 26 stand rejected under 35 U.S.C. 103(a) as being obvious over Ebrahim (U.S. Pat. No. 5,887,134) in view of Gulledge (5,644,623) in further view of Witkowski (U.S. Pat. No. 6,430,626).

(vii) Arguments

A. Rejection under 35 U.S.C. 103(a) over U.S. Pat. Nos. 5,592,622, 6,418,488, and 6,430,626

1. Claims 1-8

The Examiner rejected claims 1-8, 9-17, 28-32 as obvious based on Isfeld (5,592,522) in view of Chilton (6,418,488) in further view of Witkowski (6,430,626). Claim 1 recites a push engine to perform an unsolicited transfer of status data to a processing engine where the status data indicates whether a one of the media access devices has received packet data. The Examiner concedes that Isfeld does not describe this recited subject matter. However, neither Chilton nor Witkowski describe, suggest, or provide any motivation to modify Isfeld in a way to push such status data.

Isfeld describes a system that includes multiple IOPs (Input/Output Processors). Each IOP can have multiple MAC devices (70-1, 70-2, 70-N in FIG. 4 of Isfeld). FIG. 6 and the corresponding text identified by the Examiner illustrates sample operation of the system. In particular, FIG. 6 illustrates a packet received by IOP4 being pushed to IOP5. As emphasized in Isfeld, to reduce bus traffic, IOP5 receives a packet from IOP4

without solicitation or warning (col. 9, lines 40-41). Isfeld does not describe that IOP4 would push/perform and unsolicited transfer of MAC status data to IOP5, nor does the Examiner posit a motivation why one of skill in the art would modify the IOPs to push MAC status data about receiving packet data to other IOPs. Appellants' position remains that the Examiner's proposed continual chatter between IOPs about the arrival of packet data at each MAC would impose a considerable burden on the shared bus connecting the IOPs. Given Isfeld's goal of reducing bus traffic, Isfeld actually teaches away from the Examiner's proposed system.

Nor do either Chilton or Witkowski in any way describe or suggest pushing/performing an unsolicited transfer of such data. The passage cited by the Examiner in Chilton describes a XmtFrm register that includes bits identifying when a frame is ready for transmission from a Xmit DPR (dual port RAM) (col. 25, lines 18-59). The Examiner argues that one of skill in the art would combine Chilton with Isfeld "because if one device does not receive a type of status data (i.e., acknowledgement signal), transfer errors could accumulate in the system." (Final Office Action mailed 7/13/2005, paragraph 14). However, the relied upon passage makes no mention of the "acknowledgment signal" referred to by the Examiner. Additionally, the status data of the XmtFrm register is not "pushed" anywhere. Thus, even assuming for the sake of argument, one of skill art added an XmtFrm register to an IOP of Isfeld (presumably the Examiner's proposed combination), the resulting combination would still not teach the recited unsolicited transfer of status data.

Finally, the Examiner proposes combining Witkowksi with Isfeld and Chilton. Witkowski teaches a RX_PKT_AVAIL signal that is asserted when data is in a RX (receive) buffer. (col. 20, line 45 - col. 21, line 28). The RX_PKT_AVAIL signal, however, does not describe when a MAC has received packet data but when the packet data has already been transferred to the receive buffer from a media access device over a bus. That is, assertion of the RX_PKT_AVAIL signal indicates packet data is in a receive buffer 230, 232, while the corresponding media access device 202 may or may not currently have received packet data. Thus, Witkowski does not teach status data

Applicant : Wolrich, et. al.
Serial No. : 09/473,571
Filed : December 28, 1999
Page : 7

Attorney's Docket No.: 10559-128001
Intel Docket No.: P7867

indicating whether a media access device has received packet data. Further, the Examiner seems to propose providing a RX_PKT_AVAIL signal in an IOP formed by a combination of Isfeld and Chilton, though the Examiner's proposed combination is far from clear. However, the Examiner's proposed combination would, again, flood the bus of Isfeld with RX_PKT_AVAIL signals if such signals were pushed to other IOPs.

2. Claims 9-17, 28-32

The Examiner similarly rejected claims 9-17 and 28-32. Though including a different scope and different limitations than claim 1, claim 1 and claims 9 and 28 share a similar recitation regarding an unsolicited transfer of information on readiness of media access devices to a processing engine. As described above, Isfeld does not teach such a transfer. That is, the IOPs of Chilton do not transfer data regarding the status of the respective media access devices coupled to the IOPs. Nor would one of skill in the art burden the bus interconnecting the IOPs with a continual stream of data regarding the status of the media access devices based on either the teaching of Chilton, Witkowski, and/or the motivations provided by the Examiner for the reasons described above.

3. Claims 33-40

Claim 33 recites "multiple multi-threaded engines". Despite receiving several actions since submission of this claim, **the Examiner has not identified any aspect of the references providing this limitation.** The Examiner has, therefore, not established a prima facie case of obviousness.

B. Rejection Under 35 U.S.C. 103(a) over U.S. Pat. Nos. 5,887,134, 5,644,623, and 6,430,626

1. Claims 18-27

The Examiner rejected claim 18 as obvious based on Ebrahim (5,887,134) in view of Gulledge (5,644,623) in further view of Witkowski. Applicants, however, disagree that one of skill in the art would combine these references as argued by the Examiner nor has the Examiner provided a sufficient motivation to do so.

Claim 18 recites a router that includes a bus and parallel processor coupled to the bus. Though not explicit, the Examiner seems to rely on an SMP node of Ebrahim

(FIG. 1, 102-1) that communicates with other nodes to provide the recited parallel processor. It is unclear what the Examiner deems as the recited "bus".

Claim 18 further recites an interface to collect ready status data from media access devices coupled to the parallel processor by the us and to automatically transfer ready status data to processing engines of the parallel processor data that indicates whether a media access device has received packet data. The Examiner states that Ebrahim does not teach the recited ready status data.

The Examiner proposes modifying Ebrahim based on Gulledge. Gulledge describes a system that automatically collects historic statistics about the quality of service experienced by different handsets of a cellular phone network into a file. The file is later transferred to a computer for subsequent analysis. The Examiner argues that the combination of Ebrahim and Gulledge would provide a system that automatically transfers status data in response to data being collected.

"it would be obvious to one skilled in the art at the time the invention was made to combine Gulledge with Ebrahim because it would be faster if the status was automatically transfer once the status data collected" (Final Office Action mailed 7/13/2005, paragraph 50 on pages 11-12)

Applicants, however, do not agree that one of skill in the art would design an interface in the general purpose SMP processing node of Ebrahim solely for the purpose of an infrequent file transfer regarding cellular phone statistics. Additionally, it is unclear why the SMP node would benefit by such an interface.

Even assuming, for the sake of argument, that Ebrahim and Gulledge were somehow combined, the Examiner concedes that the resulting combination does not teach ready status data indicating whether a media access control device has received packet data. The Examiner thus proposes adding Witkowski to the combination. In particular, the Examiner relies on Witkowski's RX_PKT_AVAIL signal that is generated when a buffer has data. The Examiner seems to propose replacing the historic statistics about the quality of service experienced by different cellular handsets of Gulledge with MAC status data. The Examiner states that the motivation for the

combination would be for the same reasons as in claim 1 even though claim 1 involves two other very different references. Nevertheless, the motivation to combine Isfeld, Chilton, and Witkoswki in claim 1 was to 'to ready the packet for processing and/or transmission to other devices in the system'. However, Applicants disagree that it would have been obvious to modify a cellular handset quality measuring scheme to collect historical statistics about MACs 'to ready the packet for processing and/or transmission to other devices in the system', nor do the references provide or imply such a motivation.

(viii) Claims Appendix

1. A processor, comprising:
a module configured to collect status data from media access devices connected to a bus, the status data indicating readiness of the media access devices to participate in data transfers, the status data comprising data indicating whether a one of the media access devices has received packet data;
one or more processing engines to schedule transfers of packet data; and
a push engine to perform unsolicited transfers of the status data to the processing engines in response to the module collecting new status data.
2. The processor of claim 1, wherein the processing engines comprise:
one or more input transfer registers to receive the unsolicited transfers of status data for use to schedule the transfers of packet data.
3. The processor of claim 2, wherein the processing engines use a portion of received new status data to schedule retrievals of packet data from the devices.
4. The processor of claim 2, wherein the processing engines use a portion of the received status data to schedule transmissions of packet data.
5. The processor of claim 4, wherein the processing engines use a portion of the received status data to determine whether schedule transmissions of packet data have been completed.
6. The processor of claim 1, wherein the module is configured to poll the devices for the status data over a second bus.

7. The processor of claim 2, wherein a portion of the status data are flags indicative of whether associated devices have packet data to transmit.

8. The processor of claim 2, wherein a portion of the status data includes flags indicative of whether associated devices have space to receive packet data.

9. A method of transferring data packets over a bus, comprising:
collecting information on readiness of media access devices connected to the bus to one of transmit and receive data packets; and
transferring a portion of the collected information to a processing engine configured to schedule data transfers, the transferring being unsolicited by the processing engine.

10. The method of claim 9, further comprising:
scheduling data transfers with a portion of the devices based on the transferred portion of the collected information.

11. The method of claim 10, wherein scheduling further includes:
determining whether the transferred information is at least partly new; and
wherein the scheduling is performed in response to the transferred information being at least partly new.

12. The method of claim 10, wherein determining includes comparing a value of a time stamp transferred with the information to a previous value of the time stamp.

13. The method of claim 10, wherein scheduling further comprises:
determining whether an earlier scheduled data transfer have been completed from the transferred information.

14. The method of claim 10, wherein collecting further comprises:
polling the devices for ready status data on the availability of ports thereon; and
receiving ready status data associated with individual ones of the devices in
response to the polling.

15. The method of claim 12, wherein collecting further comprises:
writing the received ready status data to a status register; and
scheduling transfers of data packets over the bus in response to the transferred
portion of the ready status data.

16. The method of claim 9, wherein the transferred portion of the information
includes flags that indicate whether associated ports of the devices have one of space
to receive data packets and data packets ready to transmit over the bus.

17. The method of claim 16, further comprising:
polling the ports of the devices over a second bus to determine values of the
flags.

18. A router, comprising:
a bus; and
a parallel processor coupled to the bus and comprising:
a plurality of processing engines to process data transfers with a plurality
of media access devices connected to the bus; and
an interface connected to collect ready status data from the media access
devices and to automatically transfer ready status data to the processing engines
in response to the status data being collected, the ready status data indicating
readiness of the devices to participate in data transfers, the ready status data
comprising data indicating whether a one of the media access devices has
received packet data.

19. The router of claim 18, wherein the ready status data indicates the readiness of individual ones of the devices to one of receive a data packet from and transmit a data packet to the parallel processor.

20. The router of claim 18, wherein the ready status data includes a time stamp indicative of a staleness of the ready status data.

21. The router of claim 18, wherein a portion of the ready status data includes information to enable the processing engines to identify which scheduled data transfers to the devices have been completed.

22. The router of claim 18, further comprising:
a ready bus capable of transferring ready status data from the devices to the interface.

23. The router of claim 19, wherein the ready status data indicates whether associated ports of the devices are ready to perform one of a transmission of a data packet to the bus and a receive of a data packet from the bus.

24. The router of claim 20, wherein each processing engine comprises at least one input transfer register; and

the interface is configured to write ready status data to one of the input transfer registers assigned to a scheduler thread.

25. The router of claim 24, wherein the interface is configured to protect one of the input transfer registers from being read by the processing engines during the transferring of ready status data thereto.

26. The router of claim 18, wherein the devices are capable of transmitting data packets between the bus and external networks.

27. The router of claim 18, wherein the interface transfers the collected status data without being solicited to transfer the data by the processing engines.

28. An article comprising a computer-readable medium which stores executable instructions for transferring data packets over a bus, the instructions causing a processor to:

collect information on readiness of media access devices connected to the bus to one of transmit and receive data packets; and

transfer a portion of the collected information to a processing engine configured to schedule data transfers, the transferring being unsolicited by the processing engine.

29. The article of claim 28, the instructions further causing the processor to: schedule data transfers with a portion of the devices based on the transferred portion of the collected information.

30. The article of claim 29, the instructions further causing the processor to: determine whether the transferred information is at least partly new; and wherein instructions causing the processor to schedule are performed in response to determining that the transferred information being at least partly new.

31. The processor of claim 1 in which the processing engines schedule the transfer of data packets independently of the module collecting status data from the devices.

32. The processor of claim 31 in which the processing engines schedule the transfer of data packets from a device to the bus independently of the readiness of other

devices to receive the data, and schedule the transfer of data from the bus to a device independently of the readiness other devices to send the data.

33. A processor, comprising:
multiple multi-threaded programmable processing engines, individual ones of the programmable processing engines having at least one register; and
an interface operationally coupled to the multiple programmable processing engines, the interface comprising:
at least one register; and
logic to:
collect status data of at least one media access device via a bus, the status data indicating whether the at least one media access device has received packet data; and
perform a transfer, unsolicited by the programmable processing engines, of at least a portion of the collected status data stored in the at least one register of the interface to at least one register of the multiple multi-threaded programmable processing engines.

34. The processor of claim 33, wherein the at least one media access device comprises an Ethernet media access device.

35. The processor of claim 33, wherein the interface comprises a push engine, the push engine to perform the unsolicited transfer of the status data from the at least one register of the interface to the at least one register of the multiple multi-threaded programmable processing engines.

36. The processor of claim 33, wherein the interface further comprises logic to:
collect status data indicating ability of the at least one media access device to receive data to transmit; and

transfer packet data to the at least one media access device.

37. The processor of claim 33, further comprising at least one memory controller to a Synchronous Dynamic Random Access Memory (SDRAM).

38. The processor of claim 33, wherein the interface further comprises a buffer to store packet data received by the at least one media access device.

39. The processor of claim 33, wherein the at least one media access device comprises multiple media access devices.

40. The processor of claim 33, wherein the status data of multiple media access devices is stored in a single one of the at least one register of the interface.

Applicant : Wolrich, et. al.
Serial No. : 09/473,571
Filed : December 28, 1999
Page : 18

Attorney's Docket No.: 10559-128001
Intel Docket No.: P7867

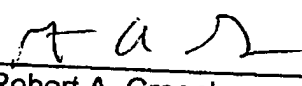
Conclusion

Given the above arguments, Applicant requests allowance of the independent claims and their corresponding dependent claims.

If any fees are due, please apply such fees to Deposit Account No. 06-1050 referencing attorney docket number: 10559-128001

Respectfully submitted,

Date: 1/13/06



Robert A. Greenberg
Reg. No. 44,133
Phone: 978-553-2060